

## Claims

What is claimed is:

- 1           1.     A method for implementing autonomic testing and verification  
2 of software fix programs comprising the steps of:  
3           receiving a software fix program; said software fix program including  
4 multiple patches;  
5           sequentially applying each patch of said multiple patches of said  
6 software fix program to a software product; and  
7           testing said software product responsive to each said sequentially  
8 applied patch.
- 1           2.     A method for implementing autonomic testing and verification  
2 of software fix programs as recited in claim 1 wherein said software fix  
3 program includes a program temporary fix (PTF); said PTF including multiple  
4 patches or programs.
- 1           3.     A method for implementing autonomic testing and verification  
2 of software fix programs as recited in claim 2 includes the step of providing  
3 an isolation manager for receiving said PTF and for sequentially applying  
4 each patch or program of said multiple patches or programs of said PTF to  
5 said software product.
- 1           4.     A method for implementing autonomic testing and verification  
2 of software fix programs as recited in claim 3 includes the step of providing a  
3 user interface coupled to said isolation manager for receiving user input  
4 selections and reporting results to the user.
- 1           5.     A method for implementing autonomic testing and verification  
2 of software fix programs as recited in claim 2 wherein the step of receiving  
3 said software fix program or said program temporary fix (PTF) includes  
4 receiving a set of PTFs, each said PTF containing multiple patches or  
5 programs; and wherein each patch or program contained in said group of  
6 PTFs is sequentially applied to said software product.

1           6.     A method for implementing autonomic testing and verification  
2 of software fix programs as recited in claim 2 includes providing an isolation  
3 manager for receiving said software fix program or said program temporary  
4 fix (PTF) and for sequentially applying each patch or program of said  
5 multiple patches or programs of said software fix program or said program  
6 temporary fix (PTF) to said software product; and wherein said isolation  
7 manager, responsive to a manual isolation user option, applies a patch or  
8 program of the multiple patches or programs of said software fix program or  
9 said program temporary fix (PTF) to said software product, notifies the user  
10 of said patch or program applied to said software product, and waits for a  
11 user option of next or done.

1           7.     A method for implementing autonomic testing and verification  
2 of software fix programs as recited in claim 6 wherein said isolation  
3 manager, responsive to receiving said next user option, applies a next patch  
4 or a next program of the multiple programs of said software fix program or  
5 said program temporary fix (PTF) to a software product, notifies the user of  
6 the next patch or next program applied to the software product, and waits for  
7 a user option of next or done.

1           8.     A method for implementing autonomic testing and verification  
2 of software fix programs as recited in claim 2 includes providing an isolation  
3 manager for receiving said software fix program or said program temporary  
4 fix (PTF) and for sequentially applying each patch or program of said  
5 multiple patches or programs of said software fix program or said program  
6 temporary fix (PTF) to said software product; and wherein a test program  
7 and expected test results are input to the isolation manager; said isolation  
8 manager sequentially applies iterations of each patch or program of the  
9 multiple patches or programs of said software fix program or said program  
10 temporary fix (PTF) and different combinations of the patches or programs to  
11 the software product and calls the test program for each applied iteration to  
12 the software product.

1           9.     A method for implementing autonomic testing and verification  
2 of software fix programs as recited in claim 8 wherein said isolation manager  
3 compares test results with the expected test results for each applied iteration  
4 and notifies the user when the test results are different from the expected  
5 test results.

1           10.    A method for implementing autonomic testing and verification  
2 of software fix programs as recited in claim 2 includes providing an isolation  
3 manager for receiving said software fix program or said program temporary  
4 fix (PTF) and for sequentially applying each patch or program of said  
5 multiple patches or programs of said software fix program or said program  
6 temporary fix (PTF) to said software product; and wherein a test program is  
7 input to the isolation manager; said isolation manager sequentially applies  
8 iterations of each patch or program of the multiple patches or programs of  
9 said software fix program or said program temporary fix (PTF) and different  
10 combinations of said patches or programs to the software product and calls  
11 the test program for each applied iteration to the software product.

1           11.    A method for implementing autonomic testing and verification  
2 of software fix programs as recited in claim 10 wherein said isolation  
3 manager saves test results in a results table for each applied iteration to the  
4 software product and displays said test results to the user.

1           12.    A method for implementing autonomic testing and verification  
2 of software fix programs as recited in claim 11 wherein said isolation  
3 manager compares all test results to each other test result to identify a  
4 problem patch or program.

1           13.    Apparatus for implementing autonomic testing and verification  
2 of software fix programs or program temporary fixes (PTFs) comprising:  
3           an isolation manager for receiving a PTF containing a plurality of  
4 programs;  
5           a user interface coupled to said isolation manager for receiving user  
6 input selections and reporting results to a user; and  
7           said isolation manager for sequentially applying each program of said  
8 multiple programs of said PTF to said software product; and for testing said  
9 software product responsive to each said sequentially applied program.

1           14.   Apparatus for implementing autonomic testing and verification  
2   of software fix programs as recited in claim 13 wherein said isolation  
3   manager is responsive to receiving a set of PTFs, for sequentially applying  
4   each program contained in said set of PTFs to said software product, and for  
5   testing said software product responsive to each said sequentially applied  
6   program.

1           15.   Apparatus for implementing autonomic testing and verification  
2   of software fix programs as recited in claim 13 wherein said isolation  
3   manager is responsive to a manual isolation user option, for applying a  
4   program of the multiple programs of the PTF to said software product, for  
5   notifying the user of the program applied to said software product, and for  
6   waiting for a user option of next or done.

1           16.   Apparatus for implementing autonomic testing and verification  
2   of software fix programs as recited in claim 15 wherein said isolation  
3   manager is responsive to receiving said next user option for applying a next  
4   program of the multiple programs of the PTF to said software product, for  
5   notifying the user of the next program applied to the software product, and  
6   for waiting for a user option of next or done.

1           17.   Apparatus for implementing autonomic testing and verification  
2   of software fix programs as recited in claim 13 wherein said isolation  
3   manager is responsive to receiving a test program; for sequentially applying  
4   iterations of each program of the multiple programs of the PTF and different  
5   combinations of the programs to the software product and for calling said  
6   test program for each applied iteration to the software product.

1           18.   Apparatus for implementing autonomic testing and verification  
2   of software fix programs as recited in claim 17 wherein said isolation  
3   manager is responsive to receiving expected test results for comparing test  
4   results with said expected test results for each applied iteration and for  
5   notifying the user when said test results are different from said expected test  
6   results.

1           19.   Apparatus for implementing autonomic testing and verification  
2 of software fix programs as recited in claim 17 wherein said isolation  
3 manager is adapted for saving test results in a results table for each applied  
4 iteration to the software product and for displaying said test results to the  
5 user.

1           20.   Apparatus for implementing autonomic testing and verification  
2 of software fix programs as recited in claim 19 wherein said isolation  
3 manager is adapted for compares all test results to each other test result to  
4 identify a problem program.

1           21.   A computer program product for implementing autonomic  
2 testing and verification of software fix programs or program temporary fixes  
3 (PTFs) in a computer system, said computer program product including  
4 instructions executed by the computer system to cause the computer system  
5 to perform the steps of:  
6           receiving a program temporary fix (PTF); said PTF including multiple  
7 programs;  
8           sequentially applying each program of said multiple programs of said  
9 PTF to a software product; and  
10          testing said software product responsive to each said sequentially  
11 applied program.